



# UNITED STATES PATENT AND TRADEMARK OFFICE

*mn*  
UNITED STATES DEPARTMENT OF COMMERCE  
United States Patent and Trademark Office  
Address: COMMISSIONER FOR PATENTS  
P.O. Box 1450  
Alexandria, Virginia 22313-1450  
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/717,671	11/21/2003	Anuj Dua	10001618-3	9068

22879 7590 06/19/2007  
HEWLETT PACKARD COMPANY  
P O BOX 272400, 3404 E. HARMONY ROAD  
INTELLECTUAL PROPERTY ADMINISTRATION  
FORT COLLINS, CO 80527-2400

EXAMINER
----------

HUISMAN, DAVID J

ART UNIT	PAPER NUMBER
----------	--------------

2183

MAIL DATE	DELIVERY MODE
-----------	---------------

06/19/2007

PAPER

**Please find below and/or attached an Office communication concerning this application or proceeding.**

The time period for reply, if any, is set in the attached communication.

# Office Action Summary

Application No.

10/717,671

Applicant(s)

DUA ET AL.

Examiner

David J. Huisman

Art Unit

2183

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

## Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

## Status

- 1) ☒ Responsive to communication(s) filed on 16 January 2007.
- 2a) ☐ This action is FINAL. 2b) ☒ This action is non-final.
- 3) ☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

## Disposition of Claims

- 4) ☒ Claim(s) 1-15 is/are pending in the application.
- 4a) Of the above claim(s) \_\_\_\_\_ is/are withdrawn from consideration.
- 5) ☒ Claim(s) 1-10 is/are allowed.
- 6) ☒ Claim(s) 11-15 is/are rejected.
- 7) ☐ Claim(s) \_\_\_\_\_ is/are objected to.
- 8) ☐ Claim(s) \_\_\_\_\_ are subject to restriction and/or election requirement.

## Application Papers

- 9) ☐ The specification is objected to by the Examiner.
- 10) ☒ The drawing(s) filed on 21 November 2003 is/are: a) ☒ accepted or b) ☐ objected to by the Examiner.  
Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).  
Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).
- 11) ☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

## Priority under 35 U.S.C. § 119

- 12) ☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).
- a) ☐ All b) ☐ Some \* c) ☐ None of:
- ☐ Certified copies of the priority documents have been received.
  - ☐ Certified copies of the priority documents have been received in Application No. \_\_\_\_\_.
  - ☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

\* See the attached detailed Office action for a list of the certified copies not received.

## Attachment(s)

- ☐ Notice of References Cited (PTO-892)
- ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
- ☐ Information Disclosure Statement(s) (PTO/SB/08)  
Paper No(s)/Mail Date \_\_\_\_\_
- ☐ Interview Summary (PTO-413)  
Paper No(s)/Mail Date. \_\_\_\_\_
- ☐ Notice of Informal Patent Application
- ☐ Other: \_\_\_\_\_

### **DETAILED ACTION**

1. Claims 1-15 have been examined.

#### ***Papers Submitted***

2. It is hereby acknowledged that the following papers have been received and placed of record in the file: RCE and Amendment as received on 1/16/2007.

#### ***Continued Examination Under 37 CFR 1.114***

3. A request for continued examination under 37 CFR 1.114, including the fee set forth in 37 CFR 1.17(e), was filed in this application after final rejection. Since this application is eligible for continued examination under 37 CFR 1.114, and the fee set forth in 37 CFR 1.17(e) has been timely paid, the finality of the previous Office action has been withdrawn pursuant to 37 CFR 1.114. Applicant's submission filed on January 16, 2007 has been entered.

#### ***Claim Rejections - 35 USC § 112***

4. The following is a quotation of the first paragraph of 35 U.S.C. 112:

The specification shall contain a written description of the invention, and of the manner and process of making and using it, in such full, clear, concise, and exact terms as to enable any person skilled in the art to which it pertains, or with which it is most nearly connected, to make and use the same and shall set forth the best mode contemplated by the inventor of carrying out his invention.

5. Claims 11-15 are rejected under 35 U.S.C. 112, first paragraph, as failing to comply with the written description requirement. The claim(s) contains subject matter which was not described in the specification in such a way as to reasonably convey to one skilled in the relevant art that the inventor(s), at the time the application was filed, had possession of the claimed

Art Unit: 2183

invention. Specifically, in claim 11, applicant claims “a fetch address queue that stores a fetch address for the line of instructions retrieved from the memory subsystem when the emulated ISA pipeline is stalled”. However, page 5, lines 30-35, of the specification, which applicant points to for support, state that “The fetch address queue 50 is used to store fetch addresses 120 sent from the x86 engine 30 when the EM pipeline 40 is stalled.” The EM pipeline, according to page 5, line 25, is not the emulation engine, but the native engine. Consequently, applicant does not have original support for showing possession of a system having a fetch address queue that stores a fetch address for the line of instructions retrieved from the memory subsystem when the emulated ISA pipeline is stalled. Applicant instead has support for a fetch address queue that stores a fetch address for the line of instructions retrieved from the memory subsystem when the native ISA pipeline is stalled. The examiner will interpret the claim language “when the emulated ISA pipeline is stalled” as --when the native ISA pipeline is stalled--.

6. Claims 12-15 are rejected under 35 U.S.C. 112, 1<sup>st</sup> paragraph, as failing to comply with the written description requirement, because claims 12-15 are dependent, either directly or indirectly, on a claim failing to comply with such a requirement.

### ***Claim Rejections - 35 USC § 103***

7. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

(a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negated by the manner in which the invention was made.

Art Unit: 2183

8. Claims 11, 12, and 14 are rejected under 35 U.S.C. 103(a) as being unpatentable over Kahle et al., U.S. Patent No. 5,732,235 (herein referred to as Kahle) and Kane et al., U.S. Patent No. 5,537,559 (herein referred to as Kane).

9. Referring to claim 11, Kahle has taught a multi-architecture computer system capable of implementing a native instruction set architecture (ISA) and an emulated ISA, wherein instructions of the native ISA are processed in a native ISA pipeline and instructions of the emulated ISA are processed in an emulated ISA pipeline (Fig.4 shows that the system is pipelined and that different hardware is used to process native and emulation instructions; therefore, it can be said that native instructions are processed by a native pipeline and emulation instructions are processed by an emulated pipeline), the system comprising:

a) a memory subsystem of the native ISA. See Fig.1 and column 2, lines 33-39.

b) a fetch engine of the native ISA, said fetch engine being electrically connected to the memory subsystem of the native ISA, wherein the fetch engine accesses the memory subsystem to retrieve a line of instructions from the memory subsystem. See Fig.1, and note that instructions are fetched into instruction queue 20. In addition, it is inherent that the native fetching unit and the memory in which native instructions are stored be electrically connected together; otherwise, it could not fetch from memory (note that the fetch engine could comprise the prefetch component 52 (Fig.2) and the buses the data cache to component 36). Finally, see column 5, lines 6-9, and note that every cycle, two instructions are fetched (i.e., the fetch bandwidth is two). These two instructions make up a line of instructions, where the line is associated with the fetch address applied to the memory in that cycle.

Art Unit: 2183

c) an engine of an emulated ISA, wherein the engine of the emulated ISA is electrically connected to the fetch engine and interfaces with the fetch engine using a handshake protocol, wherein the engine of the emulated ISA receives a line of instructions. See Fig.2 and note that the native fetch engine is connected to the emulation engine. In general, handshaking is the general communication between two components in order to complete a task. In Kahle, the fetch engine fetches instructions and sends the guest instructions to the emulation engine 36. The emulation engine will then convert the guest instructions into native instructions and send them to the instruction queue portion of the fetch engine. See column 2, lines 57-67. So, the handshaking protocol comprises the fetch engine telling the emulation engine to take action on guest instructions and the emulation engine then tells the fetch engine that the action has been taken and that processing may continue on the converted guest instructions. And, recall that a line of instructions is received. See column 5, lines 6-9.

d) Kahle has not taught that the engine of the emulated ISA receives a fetch complete signal from the fetch engine. However, Kane has taught sending a fetch complete signal to an instruction buffer (Fig.4, component 480) in the form of an exception status signal. It should be noted that this status signal arrives simultaneously with the fetched instruction, since they go hand-in-hand. See column 11, lines 60-67, of Kane. Therefore, this signal is a signal separate from the line of instructions that signifies that the fetch is complete. This exception status signal allows for a simple solution for tracking address-exceptions and generating the exceptions at the appropriate point in time (i.e., immediately upon execution). See column 3, lines 57-65, of Kane. A person of ordinary skill in the art would have recognized that such exception signals would be applicable in Kahle's system since Kahle is concerned with address exceptions. From

Fig.2 (component 54) of Kahle, it is seen that limit and attribute checks are made and exceptions are generated by the guest instructions. Therefore, it would have been obvious to one of ordinary skill in the art at the time of the invention to provide such a signal to the instruction buffer (Fig.2, component 50) in Kahle's emulation engine system which not only acts as a fetch complete signal, but also improves the efficiency of a system dealing with exceptions.

e) Kahle has not taught a fetch address queue that stores a fetch address for the line of instructions retrieved from the memory subsystem when the emulated ISA pipeline is stalled, wherein the fetch address queue is controlled by the fetch complete signal such that the fetch address is stored in the fetch address queue until the fetch complete signal is received. However, Kane has taught the concept of fetch addresses/requests being queued because of higher-priority operand fetches and other operations. See Fig.4, component 400, and column 11, lines 30-47. Furthermore, Official Notice is taken that stalling is well-known and accepted in the art. Stalling is used to correct hazardous situations in the pipeline. When stalling occurs, data in the pipeline does not advance. It would have been obvious to one of ordinary skill in the art at the time of the invention to modify Kahle to first stall the pipeline to correct any hazardous situations, and consequently, to include a fetch address queue as taught in Kane in order to store and postpone fetch requests when the pipeline is stalled because, since the fetch request cannot be processed due to data not advancing in the stalled pipeline, it must be stored somewhere. Kane has further taught that the step of storing comprises storing the fetch address in the fetch address queue until the fetch complete signal is sent. It is disclosed, in column 12, lines 11-13, of Kane, that the fetch addresses are kept in the fetch address queue until the fetch request is performed.

Consequently, the complete signal will signify that the fetch request has been performed (since it

will arrive at the instruction buffer at the same time as the fetched instruction), and as a result, the corresponding fetch request will be removed from the queue.

10. Referring to claim 12, Kahle in view of Kane has taught a computer system as described in claim 11. Kahle has further taught that the engine of the emulated ISA requests the line of instructions and the fetch engine sends the line of instructions to the engine of the emulated ISA. See Fig.2, and note that when the emulation engine decodes a branch instruction (as shown in Fig.4 and Fig.6, for instance), the branch history table is accessed, which would in turn provide a target address fetch request (if the branch is predicted taken) to the prefetch mechanism for fetching a line of instructions. This line would then be sent from the memory subsystem to the emulation engine for execution.

11. Referring to claim 14, Kahle in view of Kane has taught a computer system as described in claim 11. Kahle has further taught a macroinstruction queue that stores the instructions that were retrieved by the fetch engine. See Fig 2, component 50.

12. Claim 13 is rejected under 35 U.S.C. 103(a) as being unpatentable over Kahle in view of Kane and further in view of Whitted III et al., U.S. Patent No. 5,515,521 (herein referred to as Whitted).

13. Referring to claim 13, Kahle in view of Kane has taught a computer system as described in claim 11. Kahle in view of Kane has not taught that if a pending fetch request is canceled due to a pipeline flush, then a pending fetch request is canceled and the fetch address queue is cleared. However, Whitted has taught this type of procedure. See column 9, lines 54-67. A person of ordinary skill in the art would have recognized that in branch situations, either the



Art Unit: 2183

target address or the subsequent address would be fetched next. If the target address is to be fetched next (i.e. taken branch), then all of the subsequent pending requests that are on the non-taken path do not need to be executed. Therefore, canceling the following pending requests that are unrelated to the branch along with clearing the fetch address queue would ensure that improper fetches are discarded and rollback is avoided (rollback being the term used to describe the corrections made to values that were incorrectly updated by instructions that should not have executed). Therefore, in order to avoid fetching and executing improper instructions, it would have been obvious to one of ordinary skill in the art at the time of the invention to cancel a pending fetch request and clear the fetch address queue due to a pipeline flush.

14. Claim 15 is rejected under 35 U.S.C. 103(a) as being unpatentable over Kahle in view of Kane and further in view of Papworth et al., U.S. Patent No. 5,584,037 (herein referred to as Papworth).

15. Referring to claim 15, Kahle in view of Kane has taught a computer system as described in claim 14. Kahle in view of Kane has not taught the a speculative write pointer that prevents the macroinstruction queue from becoming oversubscribed by one or more pending fetch requests, wherein the speculative write pointer may be used to control the sending of a fetch request. However, Papworth has taught such the concept of using a speculative write pointer to track when a queue will be filled based on fetch requests. See column 2, lines 20-33. Clearly, once the queue is full, the queue will no longer accept data until some data is taken out. As a result, once the queue is full, it is inherent that fetch requests be controlled (halted), otherwise data could be incorrectly overwritten in the queue. Consequently, it would have been obvious to

Art Unit: 2183

one of ordinary skill in the art at the time of the invention to modify Kahle in view of Kane to include a speculative write pointer for tracking queue fullness and controlling the sending of fetch requests accordingly.

***Allowable Subject Matter***

16. Claims 1-10 are allowed.

***Response to Arguments***

17. Applicant's arguments filed on January 16, 2007, have been fully considered but they are not persuasive.

18. Applicant argues the novelty/rejection of claim 11 on page 6 of the remarks, in substance that:

"The applied references, particularly Kane and Kahle, individually and in combination, do not disclose all the features of amended claim 11. For example, Kane is directed to a single-pipelined computer architecture, and fetch addresses are always stored in Kane's fetch address queue. Kahle discloses a method and system for executing semantic routines in a processor that emulates guest instruction. However, Kahle does not disclose or suggest storing fetch addresses in a queue when pipelined processing stalls."

19. These arguments are not found persuasive for the following reasons:

a) The examiner asserts that the combination of the references has taught applicant's claimed invention. Kane has taught that fetch requests that cannot be performed immediately due to other operations must be queued. Other operations are known to stall and this would be a reason that a fetch request is queued (because it cannot be sent through the pipeline until the stall is cleared).

Art Unit: 2183

***Conclusion***

Any inquiry concerning this communication or earlier communications from the examiner should be directed to David J. Huisman whose telephone number is (571) 272-4168. The examiner can normally be reached on Monday-Friday (8:00-4:30).

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Eddie Chan can be reached on (571) 272-4162. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see <http://pair-direct.uspto.gov>. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

DJH  
David J. Huisman  
June 6, 2007

